6. Fa

Program Name: Fa.java **Input File: fa.dat**

In a recent thought experiment Fa conducted dealing with balancing lists of values, he imagined a list of random numbers, all non-negative within a limited range (each value less than 100). He stored them in original order in a contiguous data structure, and wanted to find the balancing point, or fulcrum, of the list. He realized that this point could either be in between two values of the list, or at an exact position of one of those values in the list, depending on the sum of the values before and after the fulcrum. He wanted the **"before"** and **"after"** sums to be equal, or as close as possible to being equal, so that the list was balanced at that point.

For example, in a list containing the values, 1 2 3 2 1, the fulcrum clearly is at position 2, right at the value 3, since the sum of the values before and after the fulcrum are equal, a sum of 3 on each side, making a perfectly balanced list.

```
1 2 3 2 1
    ^
```

He decided to indicate his findings for this list showing the **"before"** sum, the position of the fulcrum, and then the **"after"** sum, like this: **`3 ^2 3`**

For another list he tried, 3 6 3 3 5 8, he found the balancing point to be at position 3, with a sum of 12 (3+6+3) before that position, and 13 (5+8) after. He realized this wasn't a perfect balance, but he was content with "almost", since it was the best positioning of a fulcrum to best balance the list.

```
3 6 3 3 5 8
    ^
```

The report for this example was: **`12 ^3 13`**

In a third list of values, 3 3 3 3 4 3, the fulcrum was actually in between positions 2 and 3, with a sum before of 9, and after of 10.

```
3 3 3 3 4 3
    ^
```

Since the fulcrum ended up between two elements, the report format was slightly different, with the two index positions listed before and after the caret symbol, like this: **`9 2^3 10`**

Input: Several data sets, each consisting of two lines. The first line contains an integer N (2 <= N <= 50) indicating the number of elements in the list, followed on the next line by N random non-negative integers, no greater in value than 100.

Output: For each data set, show the final report formatted as shown above, with the "before" sum listed first, the fulcrum next, followed by the "after" sum, with single space separation. ***Note:*** *it is guaranteed there will be a "unique" fulcrum for each data set.*

Sample input:

```
5
1 2 3 2 1
6
3 6 3 3 5 8
6
3 3 3 3 4 3
```

Sample output:

```
3 ^2 3
12 ^3 13
9 2^3 10
```